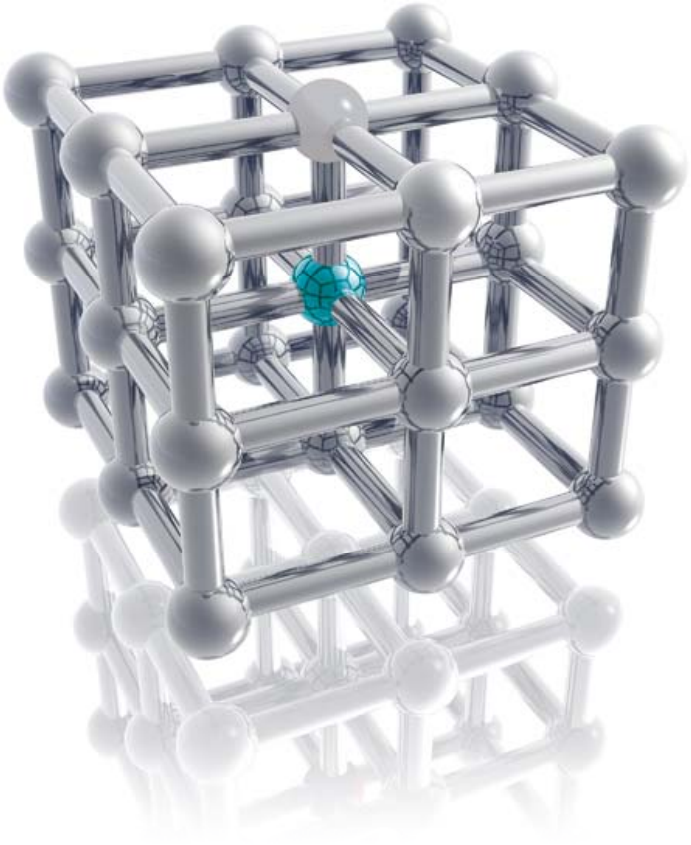


Pattern Trends

Pedit, Pselect Programs – User's Manual

Doc. No. ENP5019
Version: 28-10-2007

ASKOM® and **asix** ® are registered trademarks of ASKOM Spółka z o.o., Gliwice. Other brand names, trademarks, and registered trademarks are the property of their respective holders.

All rights reserved including the right of reproduction in whole or in part in any form. No part of this publication may be reproduced or transmitted in any form or by any means, electronic or mechanical, including photocopying, recording, or by any information storage and retrieval system, without prior written permission from the ASKOM.

ASKOM sp. z o. o. shall not be liable for any damages arising out of the use of information included in the publication content.

Copyright © 2007, ASKOM Sp. z o. o., Gliwice

ASKOM Sp. z o. o., ul. Józefa Sowińskiego 13, 44-121 Gliwice,
tel. +48 (0) 32 3018100, fax +48 (0) 32 3018101,
http://www.askom.com.pl, e-mail: office@askom.com.pl

1. Pattern Trends

asix system offers wide range of possibilities for pattern trends:

- **PEdit** program for pattern trends edition;
- possibility for automatic changing a point of locating the pattern trend beginning on the chart (pattern anchor);
- possibility for pattern anchor storage, also after **asix** restart.

ASKOM offers also solutions based on scripts, supplemented with **PSelect** program. They allow:

- user-made selection of pattern trend curve;
- automatic selection of pattern trend curve;
- transfer of chosen pattern curve to the controller.

They are treated as an additional, read-only archive type, marked with letter P.

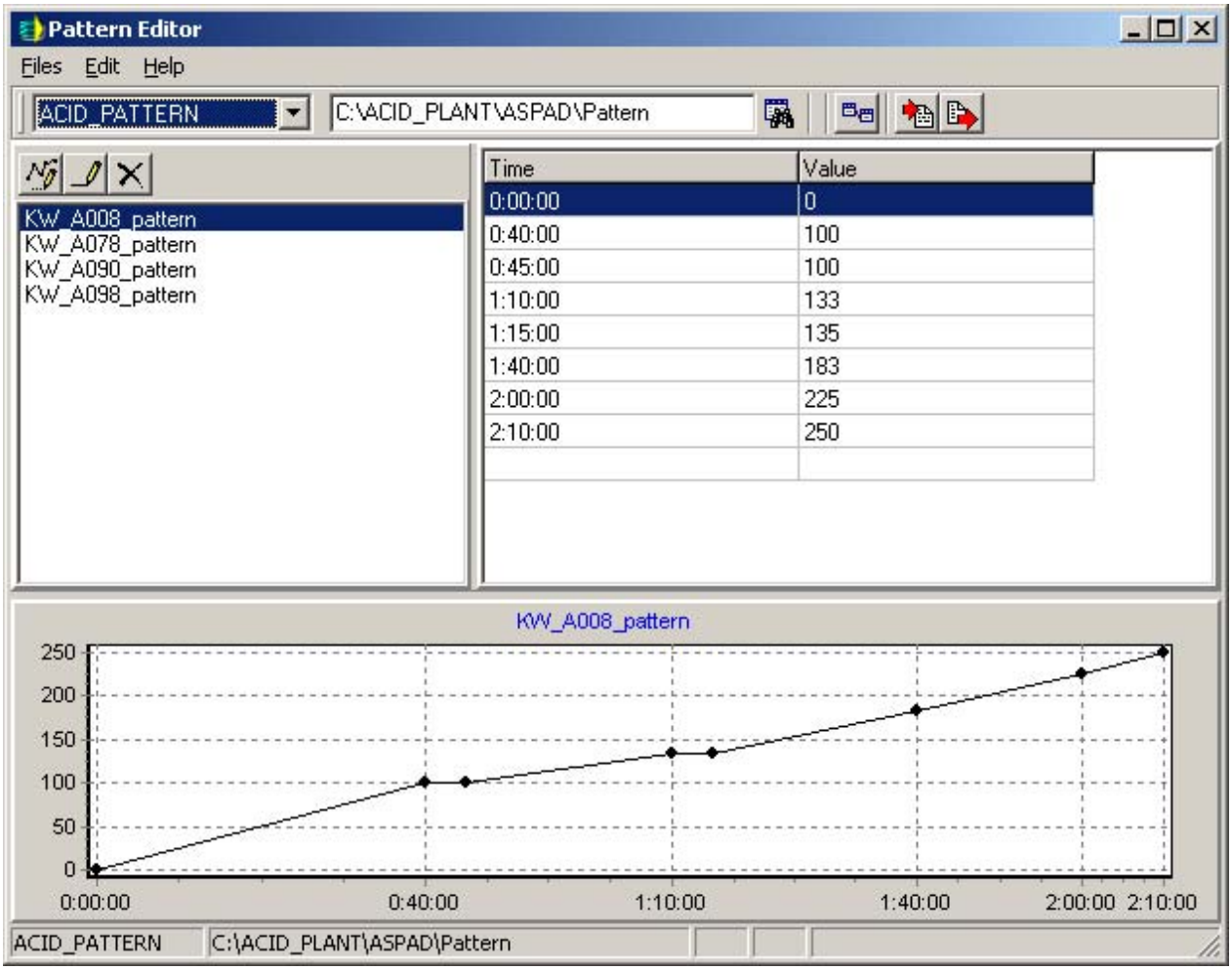
Pattern data are stored in a database analogous to B-type archive. The best solution is to place the patterns in a separate database treated as a separate ASPAD program resource. It is possible to store the B and P archives in the same database, but it's not recommended.

The patterns are prepared with use of **PEdit (**Editor of Pattern Trends**).** It is a program that enables creating database, adding and deleting patterns, editing the trend representing the pattern, and importing and exporting the pattern data to CSV text files (values separated with commas).

2. PEdit

2.1. PEdit – Pattern Trend Editor

The main window of PEdit program is shown in the following figure.



On the left a list of pattern trends from currently selected database is given. The selected pattern is represented on the right, in the form of a table and at the bottom in the form of chart.

You can zoom a part of the trend chart by its marking with a mouse. You can return to the normal displaying of trend by double-clicking anywhere in the chart area.

Indication of point on the pattern chart highligths this point in the table.

The particular program actions may be selected from menu, context menu, by clicking on the appropriate button or with keybord shortcuts.

PEdit program may be run with *–R*<re*source_name*> option, that causes BDE cache refreshing for the given resource of a pattern archive on a computer. Changes are visible on all computers within the network.

2.2. Selecting Database

You may select the database, either by selecting the name of one of the existing databases in the list on left side of toolbar of the main program window, or by selecting the directory of the database by clicking on button in main window.

If the database doesn't exist you should create it.

2.3. Creating Database

A new database may be created with use of menu action ***Files – New Database*** or by clicking on button.

Program will ask you for a name of database and its localization.

The name should correspond to the future resource name under which it will be seen in ASPAD program.

You may enter a directory path directly into the edit box or select it in the dialog window displayed by clicking on button. In the dialog window you should select the directory and accept it.

If directory doesn't exist yet, you can create it from directory selection window. In order to do it, indicate the upper level directory and click on ***New folder*** button or press ***Ins*** key. In window, which appears, enter the name of the new directory. After creating the directory you should select it in window of the directory tree and approve this selection.

In the window of directory tree you may also perform the other operations, e.g. deleting or changing the directory names.

2.4. Exporting Patterns to Text File

All patterns from opened database may be exported to a text file in the CSV format. Such files may be stored, exchanged between computers or imported to the database.

In order to export patterns to the text file, call menu action ***Files – Export CSV file*** or click on button. When the file selection window appears, select the directory where the file should be stored and enter the file name. After your selection is approved, the file with CSV extension will be generated that contains all information necessary to restore all patterns included in the currently opened database.

2.5. Importing Patterns from Text File

You may import patterns from a text file in the CSV format (values separated with commas) to the currently opened database. The CSV files have to be exported before or have format compatible to data of PGen program (former version of pattern trend generator).

In order to import the patterns from text file you should call menu action ***Files – Import CSV file*** or click on button. The file selection window appears where you should select the file from which the patterns are to be imported. After your selection is approved, the patterns will be stored in the database. The new pattern will replace any existing pattern in the database, having the same name as that of imported file.

2.6. Pattern Adding

On order to add a new pattern to the database you should perform one of the following actions:

- from main menu select ***Edit – Add***,
- from context menu of pattern list select ***Add***,
- after the list of patterns is selected press ***Ins*** key,
- click on button.

Design new pattern in the pattern-editing window and save it under new name.

2.7. Pattern Deleting

On order to delete the pattern from the database you should perform one of the following actions:

- from the main menu select ***Edit – Delete***,
- from the context menu of pattern list select ***Delete***,
- after the list of patterns is selected press ***Del*** key,
- click on button.

After approvement the pattern will be deleted.

2.8. Pattern Editing

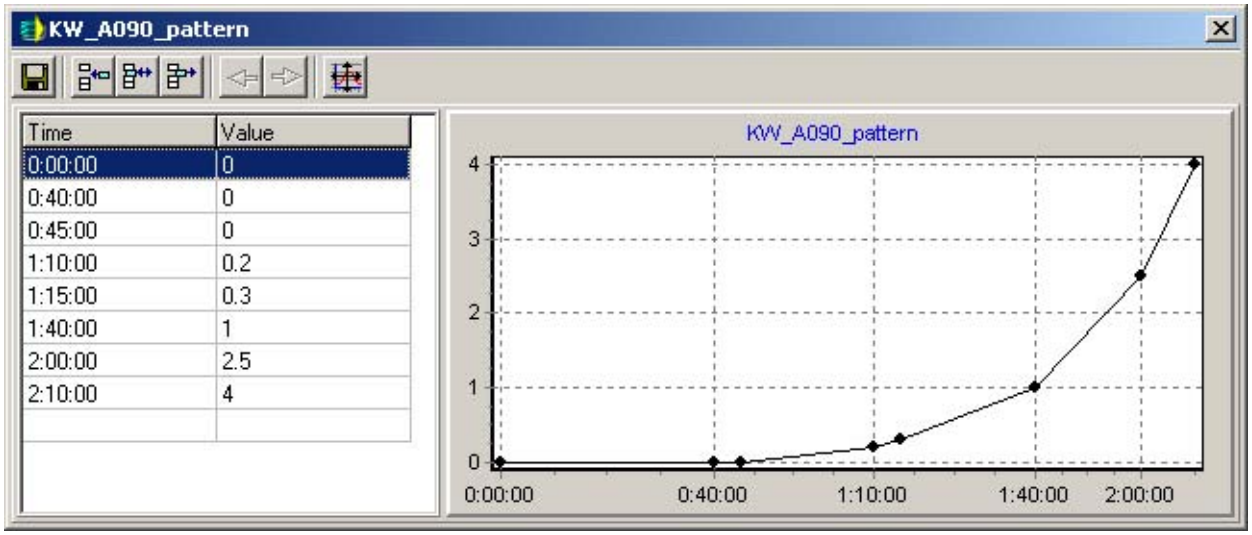
In order to modify the existing pattern you should select it on the pattern list and perform one of the following actions:

- from the main menu select ***Edit – Change,***
- from the context menu of pattern list select ***Change***,
- after the list of patterns is selected press ***Enter*** or ***Space*** key,
- click on button,
- double click on selected pattern.

In the pattern-editing window modify the pattern trend and store it under existing or a new name.
If you change the name, the new pattern will be added and the modified pattern left under old name.

2.9. Edit Window of Pattern Trends

Designing a new pattern trend or modification of existing one is performed in Edit Window illustrated in figure below.



In the window header appears the name of pattern to be modified. On the left the points of pattern in tabular form and on the right in the form of chart are given.

You can modify the pattern trend by selecting the trend point in the table or selecting it on the chart and performing one of 3 operations:

- to add new point before the selected one, press ***Ins*** key or click on button,
- to delete point press ***Del*** key or click on button,
- to modify point parameters press ***Space*** bar click on button, or double click on selected point in the table.

Changing the point parameters, when approved, deletes original modified point. The other point which time will be the same after modification as that of modified one will be also deleted.
During the adding operation, the initial values of its parameters take those of point currently selected on the list. This operation may delete the point of the pattern, if after editting the time of added point will be the same as that of the existing one.

Every operation may be cancelled by pressing ***Ctrl-Z*** keyboard shortcut or clicking on button. Such operation may be repeated by pressing ***Ctrl-Y*** keyboard shortcut or button.

You may enlarge a part of the chart by marking the rectangular area on the chart with mouse. In order to return to normal mode double-click on drawing area or press button.

Selecting point on pattern graphics is accompanied by its selection in the table.

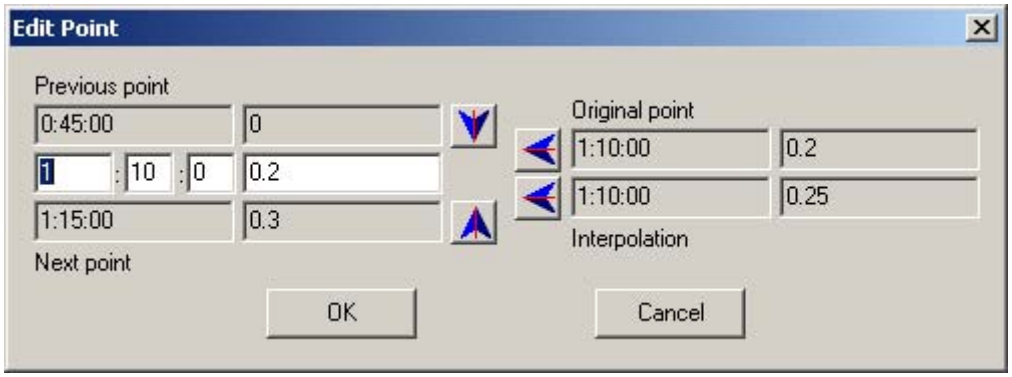
After editing is finished you can save the pattern by pressing button or accepting the write operation during the window closing. At any case program will ask you for the name under which the pattern should be saved. If you choose the old name then new pattern will replace the previous one. If you change the name, the new pattern will be saved leaving the edited pattern without any change.

The distance between succeeding points may not be larger then one day. If during saving the pattern such points are found, program will ask you for permission to generate the intermediate point at the end of every day. Lack of permission causes return to editing.

Acceptance will cause generation of additional points at the beginning of every day defined with use of linear approximation method.

2.10. Point Editing Window

You can add new points or modify existing ones in the pattern trend. These operations cause opening the point-editing window illustrated in figure below.



For edition you have 3 edit boxes to edit time of point (hours, minutes and seconds) and one edit box for editing the value of point as floating point number.

To the edit boxes designed for minutes and seconds you may enter integers from the range 0 – 59. To edit box designed for hours you may enter any positive integer, (e.g. 240 denoting 10 days). Value entered to the edit box is accepted after you have passed to edit another box.

You can fill the edit boxes by pressing one of buttons that causes inserting the value of one default point displayed with gray background. The default values are: the current value before editing and value calculated for given time from previous and next point with use of linear approximation method.

New point may be accepted by clicking on ***OK*** button or pressing ***Enter*** key or rejected by clicking on ***Cancel*** button or ***Esc*** key.

3. Defining the Pattern Anchor in the CHART Object

The elder versions of CHART object enable manual defining a pattern anchor each time when openning the CHART.

At the present moment, there is a variable of DWORD type to be used for defining the pattern anchor. The variable value has to be equal to the number of secons have passed since 1980 (this number differs 3600 seconds in the summer because time change is not taken into account).

If the variable is received from the channel, such sollution allows:

- automatic definning the pattern anchor;
- sending the information on the pattern anchor change to the controller.

The variable can also be received from NONE channel. NOTHING_LONG function should be used in such sollution. It allows to:

- define an anchor point by script and insert it to a variable controlling the pattern anchor; the pattern anchor may result from logic of a script;
- store an anchor defined in CHART object and interpret it, e.g. synchronize other patterns on other charts;
- store a variable defining the pattern anchor as a "documentation" of operator actions;
- archive an anchor with RESTORE attribute that will result in restoring the pattern anchor after system restart.

4. Ready Made Solutions for Pattern Operation

ASKOM company also hold out checked solutions for usage of pattern curves. They are based on scripts and additionally supplemented with **PSelect** application.

They consist of:

- script cooperating with PSelect application in order to allow the user-made selection of one of many patterns;
- script cooperating with PSelect application in order to allow automatic selection of one of many patterns;
- transfer of selected pattern into the driver.

4.1. Program PSelect

4.1.1 Installation

PSelect application is installed during **asix** system installation.

4.1.2 Creation of PSelect Object in the Script

After registration, PSelect application may be used in scripts. In order to use it, user has to create object PSelect.PselectObj:

```
Set PSelect = CreateObject("PSelect.PSelectObj")
```

For object created in such manner user may use two calls described below (*pattern copying, pattern selection*).

4.1.3 Copying Pattern

Method

```
PSelect.PatternCopy Resource, Source, Target
```

copies pattern of name given in *Source* variable into pattern of name given by *Target* variable, for ASPAD's resource of name given by *Resource* variable.

No variable has to be declared in **asix** system for source pattern, but the pattern has to be present in a database. A target pattern is usually used later , e.g. for displaying it on the chart, and thus it has to be declared for ASPAD as archive of P-type. A variable does not have to be declared for ASMEN, but if it is, it has to be of FLOAT type.

4.1.4 Selecting Pattern

Method

```
PSelect.PatternSelect Resource, Filter, Name
```

shows dialog box containing list of patterns with names conforming with template given by *Filter* variable and allows choice of one of them, and places its name in the *Name* variable; all these operations are performed for ASPAD's resource of name given by *Resource* variable.

User has to declare regular expression as *Filter* variable, e.g.:

```
"[0-9][0-9][0-9][0-9].*"
```

will display variables with names beginning with four digits.

Selected pattern name may be used later on, e.g. in PatternCopy method.

4.2. Examples of Scripts

There is an examplary **asix** application which uses pattern curves with support of PSelect program as well as scripts in **asix** package.

It contains two masks presenting results of PattAutSelect.vbs and PattManuSelect.vbs scripts.

Additionally, there is Pattern.vbs script – an example of usage of script and PSelect program for slection of a pattern to be sent to the driver, in main directory of the application.

4.2.1 PattAutSelect.vbs Script

From time to time, the script changes the beginning time of pattern display. It happens after execution of the following instruction:

```
TimeVar.Value = TimeBegin
```

TimeVar variable is declared as a variable controlling beginning time of pattern displayed in CHART object. *TimeBegin* is pattern beginning time calculated in the script by the following function:

```
DateDiff("s", Date1980, DateTime)
```

Additionally, *PattAut* pattern, which takes form of one of patterns stored in a database and the name of which is determined by *Pattern* variable, is occasionally displayed:

```
PSelect.PatternCopy "PattExample", Pattern, "PattAut"
```

At this place only PattAut and *TimeVar* variables have to be declared respectively for ASPAD and ASMEN.

During pattern change, *TimeVar* variable is zeroed for a second, that causes momentary turning off of the chart and its redrawing. Otherwise, the change of the pattern would not be shown on the CHART object.

4.2.2 PattManuSelect.vbs Script

Mask with a chart for this script shows the pattern with the beginning set by user. The beginnig is stored in *TimeMem* variable, value of which is restored after **asix** startup. In this way **asix** operation break does not cause change of pattern setting.

When longer break occurs, *PattAutSelect* script at the moment of startup changes the contents of *TimeMem* variable in the way the pattern would not be out of range.

There is a button on the mask used for choosing the shape of pattern progress. It runs the action that starts the *PattManuSelect* script.

During the call operation, the name of pattern archive resource and name of the pattern, to which chosen pattern should be copied, are passed. The script runs PSelect program allowing user to choose one of the patterns.

The following instruction is used for pattern selection:

```
PSelect.PatternSelect AliasName, NameFilter, SelectedName
```

And the following instruction is used for rewriting selected curve under the name given by *PatternName*:

```
PSelect.PatternCopy AliasName, SelectedName, PatternName
```

AliasName i SelectedName are received from the parameters passed to the script:

```
Set Parameters = Asix.Script.Parameters
AliasName = Parameters(1)
PatternName = Parameters(2)
```

4.2.3 Wzorzec.vbs Script

Pattern.vbs script is added to the application, but is not used by it. This script is used for sending the pattern to the driver, that may not be performed by the demo application. It may be used as basis for creation of similar script for use in a real application. The way of script operation is explained below.

ASMEN Parameterisation

Access to a data exchange region will be ensured by ASMEN by the set of appropriately declared variables which have to correspond to subsequent fields of the structure described in the chart *Driver Programming*.

- P_NBR_CURR typu INT,
- P_TIME typu LONG,
- P_NBR_PTS typu INT,
- P_DATA0 - tablica typu FLOAT,
- P_DATA1 – tablica typu FLOAT,
- …

Additionally, two variables are used for description of object to which data should be sent:

- P_OBJECT of INT type,
- P_SUBOBJ of INT type.

In *P_DATAx* arrays, values as well as times of subsequent points of a curve, will be placed. *x* stands for **subsequent** integer numbers beginning with **0**.

Because of capabilities of communication protocols, it might be impossible to send all data using singular array variable, thus it is necessary to part this table into several smaller variables. The sum of sizes of *P_DATAx* variables must be sufficient the biggest of patterns to be sent (pattern with the biggest number of points).

Some of communication protocols does not support correct sending of arrays of FLOAT type. In such case size of each variable should be set to 1.

ASPAD Parrameterisation

Dilaog choice of curve is connected with copying it into the temporary pattern, which is sent to the driver and then shared for chart display. There may be several of such temporary patterns. There must be different physical quantity, occuring in the driver, assigned to each of them.

In order to make temporary patterns available in the scripts and on the charts, they have to be declared in ASPAD module as curves of P-type, in read only resource connected with a pattern database. Names of variables for these curves does not have to be declared in ASMEN program database, however if we want to declare them they must be of FLOAT type.

Original curves, from which we make the selection does not have to be variables of ASPAD or ASMEN.

Definition of Pattern Curves

New pattern curves with names of the following form should be declared in a database:

Xxxxname

where:

xxxx	- is a 4-digit number (e.g. 0001);
name	- is an arbitrary text containing max 11 signs.

PEdit application should be used for pattern curve defining.

Declaration of Operator Action

Sending of pattern to the driver is performed by **Pattern.vbs** script, which is ran using the

following operator action:

SCRIPT, Pattern, *Resource PatternName Ob Podob* **//IT:0 //Watek:PSelect //1Plik**

Resource is a name of ASPAD resource.

PatternName is a name of pattern curve to be sent - into which a chosen curve will be copied.

Ob and *Podob* are numbers describing which technological object or physical quantity the operation will apply with. Meaning of these digits should be agreed with a developer of controller software.

Pattern.vbs script have to be placed in the **asix** mask path.

Driver Software

The standard data format shown in the table below is used for pattern storage in the controller:

Bite Number	**Variable type**	**Parameter**	**Description**
0	INT	NBR_CUR	Nr of curve (synchronization >< 0), sent as the last one
2	DINT	TIME	Time basis in sec., for example: 3600 for 1 hour scale
6	INT	NBR_PTS	Number of curve points
8	REAL	Y_0	Y coordinate point. 0
12	REAL	X_0	X coordinate point. 0
16	REAL	Y_1	Y coordinate point. 1
20	REAL	X_2	X coordinate point. 1
8n+8	REAL	Y_n	Y coordinate point. N
8n+12	REAL	X_n	X coordinate point. N

All pattern curves are transmitted through common region of data exchange.

Driver software tracks a value of first word of this region and when the value differs from zero, basing on this value the software identifies the pattern (stored in the region) and copies it to the driver arrays. After a copying operation the value of first word is set to zero.

On data sending side, data may only be sent if the value of first word id data share region is equal to zero.

Beside the region described above, two additional INT type numbers describing a technological object or physical quantity (which a sending operation applies to) should be prepared.

Index

Copying Pattern, 11
Creating Database, 4
Creation of PSelect Object in the Script, 11
Defining the Pattern Anchor in the CHART Object, 9
Edit Window of Pattern Trends, 5
Examples of Scripts, 12
Exporting Patterns to Text File, 4
Importing Patterns from Text File, 4
Installation, 11
PattAutSelect.vbs Script, 12
Pattern Adding, 5
Pattern Deleting, 5
Pattern Editing, 5
Pattern Editing Window, 7
Pattern Trends, 1
PattManuSelect.vbs Script, 13
PEdit – Editor of Pattern Trends, 3
Program PSelect, 11
Ready Made Solutions for Pattern Operation, 11
Selecting Pattern, 12
Wzorzec.vbs Script, 13

1. PATTERN TRENDS 1

2. PEDIT 3

2.1. PEdit – Pattern Trend Editor 3
2.2. Selecting Database 4
2.3. Creating Database 4
2.4. Exporting Patterns to Text File 4
2.5. Importing Patterns from Text File 4
2.6. Pattern Adding 5
2.7. Pattern Deleting 5
2.8. Pattern Editing 5
2.9. Edit Window of Pattern Trends 5
2.10. Point Editing Window 7

3. DEFINING THE PATTERN ANCHOR IN THE CHART OBJECT 9

4. READY MADE SOLUTIONS FOR PATTERN OPERATION 11

4.1. Program PSelect 11
4.1.1 Installation 11
4.1.2 Creation of PSelect Object in the Script 11
4.1.3 Copying Pattern 11
4.1.4 Selecting Pattern 12
4.2. Examples of Scripts 12
4.2.1 PattAutSelect.vbs Script 12
4.2.2 PattManuSelect.vbs Script 13
4.2.3 Wzorzec.vbs Script 13